IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-5, May 2019 E-ISSN: 2347-2693

# Offline Handwritten Character Recognition using Neural Networks

**Hemant Yadav[1*], Sapna Jain[2]**

[1] Department of Computer Science, MRK Institute of Engineering and Technology, Saharanwas, Rewari, India
[2] Department of Computer Science, MRK Institute of Engineering and Technology, Saharanwas, Rewari, India

[*]*Corresponding Author: yadavhemantyadav@gmail.com, Tel.: +91-98182-63577*

DOI: https://doi.org/10.26438/ijcse/v7i5.838845 | Available online at: www.ijcseonline.org

Accepted: 10/May/2019, Published: 31/May/2019

***Abstract*—** Handwritten character recognition is currently a under research field. A lot research is getting done in this field in which the point of interest is to receive as higher accuracy as possible in a distorted writing. That is as we know the way of writing of different person is different, so to recognize every writing with a greater accuracy is the point of concern. In this paper we proposed a method for different languages handwritten character recognition. The main focus is to train the model with pre-set data and then using that trained model to test the handwritten character passed to it. In our proposed method we used MATLAB to design our code, in this the model can be trained on runtime also.

***Keywords*—** Handwritten Character, Character Recognition, Feature Extraction, Neural Networks, Image Recognition, Offline Character Recognition.

## I. INTRODUCTION

Handwriting is an age-old and elegant way of communication and is a distinctive characteristic of an individual. The writing style of a person depends on several factors such as the writing medium, state of mind, environment, mood of the individual, etc. Obtaining high accuracy in offline handwritten character recognition is a challenging task. Several factors like background noise, different writing style of the writers, variations in character size, pen width, pen ink, character spacing, skew and slant, similarity of some characters in shape and size, influence the character recognition rate. Other significant factors, for instance, the absence of header line caused by the writer or segmentation of modifiers and touching characters also become substantial in design an efficient offline handwritten character recognition system. Generally, offline handwritten recognition is the scanned pictures of pre-written textual material on paper, and online handwritten identification are the throughout writing activity on a specially designed pen in an electronic device. Recognition of offline handwritten documents has been a vital field for research in the broad domain of the pattern recognition. Over the past few years, various laboratories all over the world showed their intense involvement studies on handwriting recognition [1]. OHR offline handwritten recognitions embraces the transformation of an image comprising a handwritten text into a format which is understandable by the computer. The text on the picture is considered as an unchanging depiction of handwriting [2]. The literature has reported different explanations for the recognition of characters in Indian calligraphy. It is necessary to mention that in the context of Indian scripts' character recognition, most of the published works deal with pre-printed papers, but there are only rare research papers representing handwritten character recognition. In recent past, several contemporary surveys of the existing techniques in the field of OHR in Indian regional calligraphy were performed to support the researchers in the subcontinent, and hence a straight attempt is made in this research to discuss its advancement. Handwriting recognition is a growing optical character recognition (OCR) technology that has to be exceptionally robust and adaptive. OHR is a widely researched and a well-known problem which has been studied to a certain extent and utilised to solve problems for a few applications, such as handwriting recognition in personal checks, recognition of hand-printed text in application forms, postal envelope, and parcel address readers [3]. Handwritten character recognition is a standout amongst the most inspiring and appealing territories of pattern recognition [4], [5]. It is characterised as the way towards detecting segments, distinguishing characters and symbols from the scanned image.

In this process, characters from the input images are identified and converted into UNICODE or similar machine-editable text [6], [7], [8]. Character recognition contributes incredibly towards the advance of automation method and enhances the interface amongst the human and machine in several applications.

For the last few decades, character recognition is getting additional importance because of the broad domain covering applications. An explicit identification of handwritten typescripts along with their transformation into device editable forms is a crucial and hard research area within the

 

domain of pattern recognition. It builds up the need to protect original manuscripts and records of ancient significance. The process to perform recognition in a system for characters is to perform the following steps in sequential order as follows Image acquisition, Image preprocessing, Segmentation, feature extraction, classification and finally post processing [9]. Section I contains the introduction of Handwriting character recognition. Section II contain the related work in Offline Handwritten Character Recognition, Section III contain the work proposed, Section IV contain the results achieved from the proposed work, section V gives conclusion on the work proposed.

## II. RELATED WORK

Handwritten character recognition is a field of vast approaches. Many researchers have worked in it and still work going on. In handwritten character recognition, many different machine learning approaches are used like neural networks, deep learning, etc. But no single algorithm is derived with maximum accuracy to detect the handwritten character of many languages if not all. And one of the most critical aspects is that all languages are still not supported for handwritten character recognition. Different researchers have used various methods on different styles to achieve greater accuracy in handwritten character recognition.

The highest accuracy is found on digits, and English alphabets as these are the most commonly used languages to test the algorithms designed for handwritten character recognition. The most common techniques to do so are deep learning
and neural networks. But recently researchers have started to combine these or use some other methods too for getting better results or for having broader data space to work with. In a recent paper, the researcher used Bidirectional Long Short-term memory networks to recognise printed Ethiopic script, in which they got an average character error rate to be 2.12% [1]. In this, they used Ethiopic Script as it contained a large number of characters from different languages. They trained their model with varying techniques of degradation. Similarly, in another paper we find the usage of neural networks to recognise handwritten alphabets. In the particular article, the researchers used gradient feature extraction and geometric feature extraction instead using back propagation network for testing and training the data.[2]

But one of the common problems in these papers was the need of extensive training data on the basis of which the machine learns that a single character can be written in how many different ways so that recognition can be done more accurately. But generating such a broad set of training data is not possible so quickly and embedding such a large collection of data for different languages will slow down the speed of the system. Researchers did recognise this problem by trying to generate the training data also by means of the computer. That is a single data set will be processed with different distortion and degradation techniques to create many other training data [3]. Some of the used methods were Elastic distortion, Spline transformation, Background blending, etc.

In another approach, researchers tried to combine two well-known methods Support vector machine and convolutional neural networks [4]. In which they provided features extracted by a convolutional neural network as input to support vector machine. But in most of the above-stated techniques the language used to test their accuracy and working was either digits or English alphabets. Not much work is conducted on other different languages which is a topic of research. As in the case of optical character recognition of printed material we have seen that not all algorithms yield the same result on different languages. Same is applied in the scenario of handwritten character recognition.[5] To achieve greater accuracy in identification of different alphabets and words in English language and on digits researchers have tried a lot of methods or combination of methods. But now we should apply those on different languages to find out whether they provide the same
accuracy for other languages too.

The inherent inconsistency found in the writing style of an individual is a significant challenge in recognition of the handwritten characters. This section presents an extensive literature survey of character recognition methods. In recent past, several ways such as loop aspect ratio [10], pixel distribution and upper lower profile, width, height, area, density, aspect ratio and separator length between two successive connected characters [11], bi-level co-occurrence [12], run-length histogram [13], connected component profiles, analysis [14], Gabor filters [15], word physical sizes, crossing count histogram, baseline profile features, overlapping areas [16] and steerable pyramids transform [17] have been used to determine statistical characteristics. The character recognition assignment has been a testing venture through different methodologies for instance template matching, analytical procedures, Hidden Markov Models (HMMs), Bayesian, Neural Network and so forth. [17] reported a method to recognise the characters of the Devanagari script in two steps. In the first step, strokes are identified while in a second step the character recognition takes place based on the strokes identified in the preliminary step. [18] have proposed Bangla handwritten character recognition in the multi-stage classifier. Here, the first stage consisted of the extraction of high-level attributes and core level classification, and further, the removal of low-level features assisted the final ranking in the second stage. [19] presented a Generalized Hausdorff Image Comparison (GHIC) system for Devanagari script recognition, which is a form of template matching method for overcoming certain disadvantages of the traditional template matching approach. [20] have proposed to recognise Bangla numerals for sorting the postal documents for the Indian postal automation system

 

using a two-stage Multilayer Perceptron (MLP) based classifier and Modified Quadratic Discriminative Function (MQDF) classification for Bangla handwritten word recognition. They considered contour pixels in a character image and obtained a histogram based features in a directional chain code and demonstrated the applicability of their work in the Indian Postal Automation context. [21] have implemented PCA for Tamil online handwritten character recognition. The quadratic interpolation features based classifier is proposed by [22] for offline handwritten Gurmukhi characters and numeral recognition. They emphasised on the contour points of the characters for extracting the features directional chain code information. Several blocks were created for separating the whole set of characters, and then chain code histogram was produced for every block. A quadratic interpolation based classifier employing 64-dimensional attribute along with chain code features have been used for classification. An MLP classifier for Bangla handwritten character recognition is proposed by [23], features are calculated by computing local chain-code histograms of input character images and calculated for the contour as the representation of the skeletal input character images. [24] have proposed a string matching algorithm for improving the performance of the recognition system. In [25], authors have proposed MQDF classifier for offline Bangle handwritten compound character recognition system. In this system, the features used primarily lay on directional information bestowed from the arctangent of the gradient. [26] presented an elastic matching technique in two stages for the recognition of online handwritten Gurmukhi script. For designing handwritten Devanagari OCR, [27] have employed feed-forward MLP in one hidden layer and trained in backpropagation algorithm in neural network and Gradient features. [Sharma and Jhajj 2010] have suggested a handwritten Gurmukhi character recognition system using zoning density based features. [28] have employed multi-layer perceptron (MLP) and SVM classifier for Bangla handwritten core and compound character recognition and Confusion matrix has been computed to recognise the results of the MLP classifier. The combination of different features and classifier have been presented by [29] for offline handwritten Gurmukhi character recognition. [30] showed recognition of Devanagari numerals utilising deep learning of Artificial neural networks in Histogram of Oriented Gradient (HOG) features. [31] proposed Fringe Distance Map, the histogram of oriented and shape descriptor to identify nearly 250 classes including middle, upper and lower zone symbols as well as conjunct characters, achieved an accuracy of approximately 98% on 16,000 symbols database using multiple kernel learning-based SVM classifier.

## III. PROPOSED WORK

An offline handwritten character recognition system has five stages namely image acquisition, pre-processing, segmentation, feature extraction and finally classification. In the present research for handwritten character recognition, the work is confined to isolated characters only thus eliminating the need of the segmentation phase.

### *1. Image Acquisition*

This is the preliminary phase of an off-line handwritten recognition system. This study is undertaken by investigating the samples of handwritten characters that are collected from different writers. In this proposed system each aspect of each language has 55 examples.

### *2. Pre-processing*

An image is a two-dimensional array of picture elements denoted by binary values arranged in rows and columns in the format of 0 and 1. In an 8-bit image of gray scale, each pixel or picture element is of a fixed intensity whose values range from 0 to 255 in binary format. A gray scale image has variation of only two values Black and White that means all the intensities generated in a gray scale image is a potential value between black and white. A normal image categorised into a gray level image has a colour depth of 8 bit that is each colour depicts 8 bit binary values. Whereas in a true colour image the colour depth increases to 24 bits that is each colour is represented by 24 bits. So, if we calculate according to following parameters, we have 16 million colour representation in a true colour image. In some cases, the gray level image is also depicted by a 16 bit colour depth system which leads to 65536 variation of gray scale. There are two general groups of images: vector graphics (or line art) and bitmaps (pixel-based or images).

**Thinning:**

Thinning is an operation in which foreground values or pixels are removed from an image. These pixels are selected values. By removing these pixels, the connectivity and topology of the image is not broken, it is preserved. That is original image or region is preserved while throwing foreground pixels which do not have a heavy effect on the image. Thinning can be defined as a morphological operation or process.

### *3. Segmentation*

Segmentation is the process in which the lines are traced in such a way that a text can be extracted from the image. That is in short image conversion to text. For this first of all the image is cleaned so the noisy lines can be removed so that text tracing can be done easily and precisely. Then pixels or values which are restriction in making of the binary image are removed. Then that particular image which has only

 

binary image properties and binary image values are stored in memory. This step makes the complete process faster.

- Divide the text into rows
- Divide the rows into words
- Divide the word into letters

*4. Feature Extraction*

In image processing for the purpose of pattern recognition or character recognition, feature extraction is a much-needed feature. Feature extraction removes unwanted dimensions from the binary image.

*5. Classification*

The algorithm used for classification is artificial neural networks. The input values or number of features is directly associated with the features of the text to be recognised. For e.g. If we take set of English alphabets then we must consider number of outputs to be 27 as there are 26 alphabets and 1 reject neuron.

## IV. EXPERIMENTAL RESULTS

The present section deals with the algorithm and results obtained through a simulation study.

### Image Processing

Pre-processing is a step in which we manipulate the image to be recognised in a segmented image suitable for recognition. In this the input image is converted into a gray scale image. Image transform into binary image that means in the form of black in white image.

### Grey Scale Conversion

As each colour pixel is described by a triplet (R, G, B) of intensities for red, green and blue colour. We can map that to a single number giving a grey scale value. There are many approaches to convert the colour image into a grey scale. Here the common method is used for the colour to grey scale conversion.

**Algorithm: Gray** Scale Conversion
**Input: Scanned Image**
**Output: Gray** Scaled Recognized Image
**Step 1:** Begin
**Step 2:** Give input image.
**Step 3:** Run continuously from x=0 to image's width
**Step 4:** Run continuously from y=0 to image's height
**Step 5:** Determine RGB value in the form of RGB(i,j)
Set int col = in Pixels[x][y];
Set int r = col & 0xff;
Set int g = (col >> 8) & 0xff;
Set int b = (col >> 16) & 0xff;
Set int gs = (r + g + b)/3;
**Step 6:** Place pixel values:
Set in Pixels[x][y] = (gs | (gs << 8) | (gs << 16));
Set gimage. setRGB (x, y, in Pixels[x][y]);
**Step 7:** Display Computed Image.
**Step 8:** End

### Binarization

This process converts a gray level image of 256 values into a binary image with only black and white values. The easiest way to perform this task is to set a threshold value for the pixel intensities, in which we define the values to be black for higher intensity and white for lower intensity as compared to threshold value.

**Algorithm: Image** Binarization (Thresholding)
**Input: Gray** Scaled Computed Image
**Output: Black** and White Image
**Step 1:** Begin
**Step 2:** Take input gray scaled computed image.
**Step 3:** Run continuously from x=0 to image's width.
**Step 4:** Run continuously from y=0 to image's height.
**Step 5:** Place threshold marker.
**Step 6:** Determine RGB value in form of RGB (i, j)
Set int col = in Pixels[x][y];
Set int r = col & 0xff;
Set int g = (col >> 8) & 0xff;
Set int b = (col >> 16) & 0xff;
Set int gs = (r + g + b) / 3;
**Step 7:** If pixel(gs) is Above Threshold Then
*{*
Set r=g=b=0;
*}*
else
*{*
Set r=g=b=255;
*}*
**Step 8:** Place pixel values as:
Set in Pixels[x][y] = (b | (g << 8) | (r << 16));
Set bimage. setRGB (x, y, in Pixels[x][y]);
**Step 9:** Display computed binary image.
**Step 10:** End

### Segmentation

This process is performed once the image is pre-processed and is ready to get segmented for further procedure. Segmenting the image means conversion or division of the image into lines, then further dividing lines into words which at last are converted into characters. Then

 

these extracted characters are used to extract features which help in performing recognition. Segmentation is a necessary step to be executed before feature extraction. The involved steps are as follows:

- Line Segmentation
- Word Segmentation
- Character Segmentation

**Line Segmentation**

In this particular process a horizontal scan is performed on the values of pixels. The values of pixels or a particular line can be identified by finding out a particular scan where you never encounter a black pixel.

**Line Segmentation Algorithm:**
**Input:** Binarized Image Document
**Output:** Segmented lines from Document Image
**Step 1:** Initialize.
**Step 2:** Insert the previously computed image.
**Step 3:** Run continuously from x = 0 to Image's Height.
**Step 4:** Run continuously from y = 0 to Image's Width.
**Step 5:** Scan horizontal pixel row.
**Step 6:** Determine RGB value in the form of Pixels[x][y]
**Step 7:** Pixel values with no black pixel found then

*The {*Segment that line from the image

*}*
**Step 8:** End

**Word Segmentation**

This process is performed by scanning the pixel values vertically. We start from the left and scan a whole vertical column consisting of different pixel values. The separation between different words is encountered by performing a scan where no black pixel is found then the value predefined in the columns.

**Word Segmentation Algorithm:**
**Input:** Lines segmented from previous output and avgpxl = average pixel width for word separation
**Output:** Segmented words from the line
**Step 1:** Initialize
**Step 2:** Insert the last output image.
**Step 3:** Run continuously from x = 0 to Segmented Height.
**Step 4:** Run continuously from y = 0 to Segmented Width.
**Step 5:** Scan pixel values vertically.
**Step 6:** Determine RGB value in form of in Pixels[x][y]
**Step 7:** Pixel values with no black pixel is found for the value of avgpxl or more then that then

*{*
Segment Word from lines
*}*
**Step 8:** End

**Character Segmentation**

This process is performed in similar manner as word segmentation. In this also the procedure is same to start from the left and scan vertically. We scan all pixel values vertically and separation is encountered if there is no black pixel in the column sequences.

**Character Segmentation Algorithm:**
**Input:** Words segmented from lines
**Output:** Characters segmented from words
**Step 1:** Begin
**Step 2:** Insert the output of last step.
**Step 3:** Run continuously from x = 0 to Section height.
**Step 4:** Run continuously from y = 0 to Section width.
**Step 5:** Scan Vertical pixel column.
**Step 6:** Determine RGB value in form of in Pixels[x][y]
**Step 7:** Pixel value with no Black pixel is found then

*{*Characters to be segmented from word.

*}*
**Step 8:** End

**Feature Extraction**

As individual characters have been separated, character image can be resized to 15 x 20 pixels. If the features are extracted accurately then the accuracy of recognition is more. Here we have used the 15 x 20 means 300 pixels as it is for feature vector. This extracted feature is stored in the .dat file

## Results

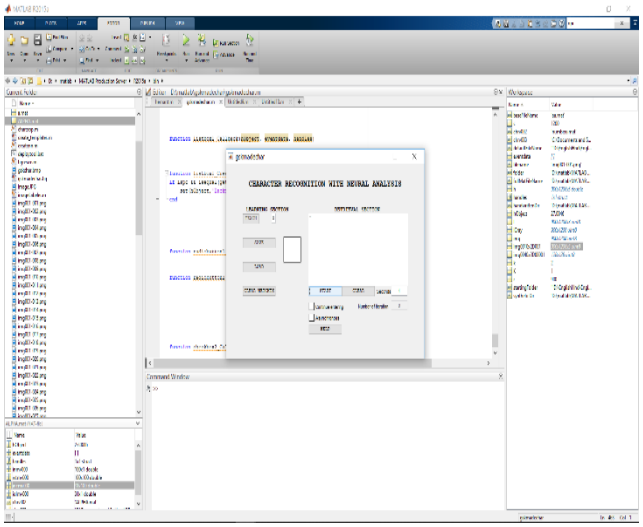


Fig 1

 

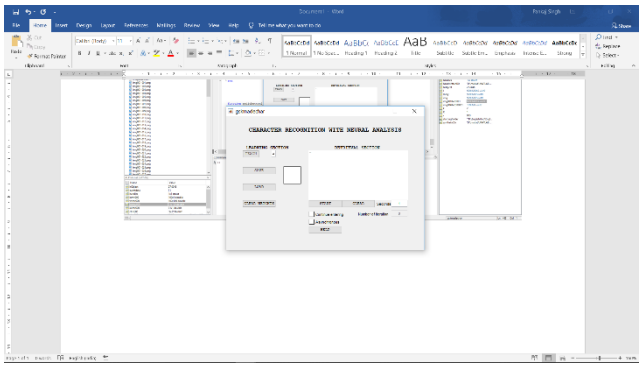

Fig 2

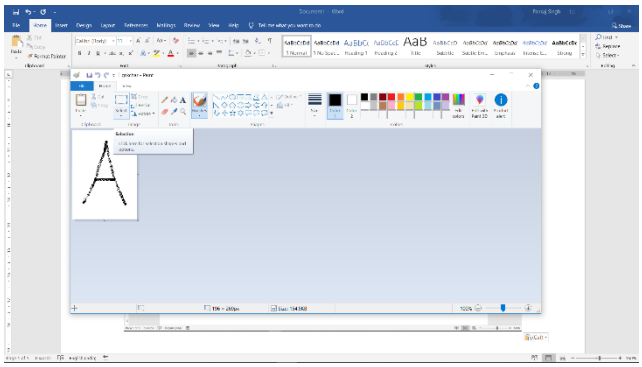

Fig 3

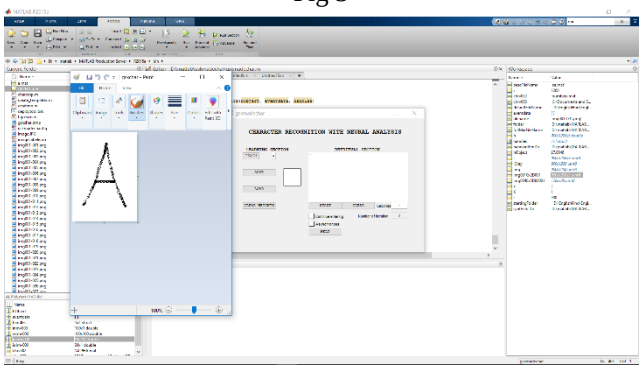

Fig 4

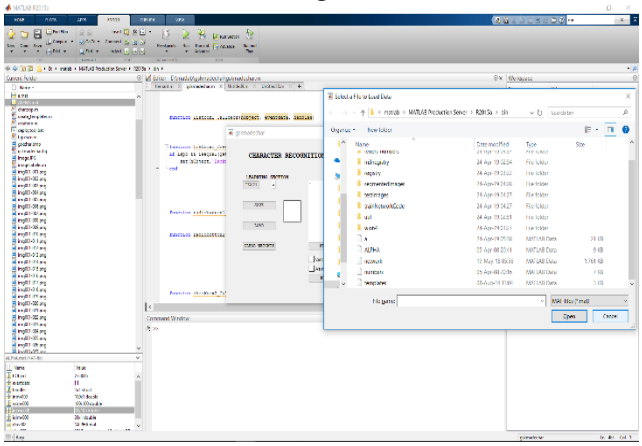

Fig 5

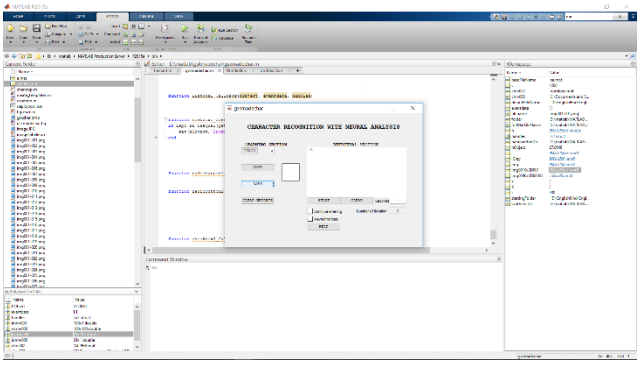

Fig 6

The proposed work shows that once the system is trained with 55 training image the accuracy achieved is much better. Proposed work also shows that if the system is trained at run time also then also the accuracy of the system remains same as the previous way of the training. But the results also tell that using multiple languages or recognizing multiple languages at the same time is still a difficult task and different languages have to be recognized separately.

## V. CONCLUSION

Many regional languages throughout the world have different writing styles which can be recognised with HCR systems using proper algorithm and strategies. We have to learn for the recognition of English characters. It has been found that recognition of handwritten character becomes difficult due to the presence of odd characters or similarity in shapes for multiple characters. The scanned image is pre-processed to get a cleaner image, and the characters are isolated into individual characters.

Pre-processing work is done in which normalisation, filtration is performed using processing steps which produce noise free and clean output. Managing our evolution algorithm with proper training, evaluation another step wise process will lead to the successful output of system with better efficiency. Use of some statistical features and geometric features through the neural network will provide better recognition result of English characters. This work will explore more possibilities for the future work in combining different languages at a single platform. This proposed work will help in creating OHCR for other languages.

## REFERENCES

[1] M. Kumar, M. K. Jindal, and R. K. Sharma, *"Classification of characters and grading writers in offline handwritten Gurmukhi script,"* Proc. Int. Conf. Image Inf. Process., ICIIP 2011.

[2] U. Pal, R. Jayadevan, and N. Sharma, *"Handwriting Recognition in Indian Regional Scripts,"* ACM Trans. Asian Lang. Inf. Process., 2012.

 

[3] E. Kavallieratou and S. Stamatatos, *"Discrimination of machine-printed from handwritten text using simple structural characteristics,"* Proc. 17th Int. Conf. Pattern Recognition, Vol.1, p. 437–440, ICPR 2004.

[4] A. K. Jain, *"Data clustering: 50 years beyond K-means,"* Pattern Recognit. Lett., vol. 31, no. 8, pp. 651– 666, 2010.

[5] M. Kumar, M. K. Jindal, and R. K. Sharma, *"Review onOCR for handwritten indian scripts character recognition,"* Commun. Comput. Inf. Sci., vol. 205 CCIS, pp. 268–276, 2011.

[6] K. Singh Siddharth, M. Jangid, R. Dhir, and R. Rani,*"Handwritten Gurmukhi Character Recognition Using Statistical and Background Directional DistributionFeatures."* International Journal on Computer Science and Engineering (IJCSE) 3, no. 06, 2011.

[7] D. Sharma and P. Jhajj, "*Recognition of IsolatedHandwritten Characters of Gurumukhi Script usingNeocognitron,"* Int. J. Comput. Appl., vol. 4, no. 8, pp. 9–17, 2010.

[8] Due Trier, A. K. Jain, and T. Taxt, *"Feature extraction methods for character recognition-A survey,"* Pattern Recognit., vol. 29, no. 4, pp. 641–662, 1996.

[9] M. K. Mahto, K. Bhatia, and R. K. Sharma, *"CombinedHorizontal and Vertical Projection Feature ExtractionTechnique for Gurmukhi Handwritten Character Recognition,"* In International Conference on Advances in Computer Engineering and Applications (ICACEA),pp. 59–65, 2015.

[10] C. L. Liu and C. Y. Suen, *"A new benchmark on the recognition of handwritten Bangla and Farsi numeral characters,"* Pattern Recognit., vol. 42, no. 12, pp. 3287–3295, 2009.

[11] H. Ma and D. Doermann, "*Word Level ScriptIdentification for Scanned Document Images,"* SPIE Conf. Doc. Recognit. Retr., pp. 124–135, 2004.

[12] Zheng, Y., Liu, C. and Ding, X., "Single-character type identification." In Document Recognition and Retrieval IX, Vol. 4670, pp. 49-57, December 2001.

[13] Zhou, L., Lu, Y., & Tan, C. L. *"Bangla/English script identification based on analysis of connected component profiles."* In International Workshop on Document Analysis Systems, pp. 243-254, 2006.

[14] S. Haboubi, S. Maddouri, N. Ellouze, *"Diff˜erenciation de documents textes Arabe et Latin par filtre de Gabor"*, 2007. wwd

[15] S. Mozaffari and P. Bahar, *"Farsi/arabic handwritten from machine-printed words discrimination,"* In Proceedings - International Workshop on Frontiers in Handwriting Recognition, IWFHR, 2012.

[16] G. Muhammad, M. H. Al-Hammadi, M. Hussain, and G. Bebis, *"Image forgery detection using steerable pyramid transform and local binary pattern,"* Mach. Vis. Appl., vol. 25, no. 4, pp. 985–995, 2014.

[17] S. D. Connell and A. K. Jain, *"Recognition ofUnconstrained On-Line Devanagari Characters,"* ICPR,1–4, 2000.

[18] A. F. R. Rahman, R. Rahman, and M. C. Fairhurst, *"Recognition of handwritten Bengali characters: A novel multistage approach,"* Pattern Recognit., *35*(5), 997-1006, 2002.

[19] H. Ma and D. Doermann, *"Adaptive Hindi OCR Using Generalized Hausdorff Image Comparison",* ACM Transactions on Asian Language Information Processing (TALIP), 2(3), 193-218, 2003.

[20] U. Pal, K. Roy, and F. Kimura, *"A Lexicon-Driven Handwritten City-Name Recognition Scheme for Indian Postal Automation,"* IEICE transactions on information and systems, no. 5, pp. 1146–1158, 2009.

[21] V. Deepu, S. Madhvanath, and A. G. Ramakrishnan,*"Principal component analysis for online handwritten character recognition,"* Pattern Recognit. (ICPR), 17th Int. Conf., vol. 2, no. 2, p. 327–330 Vol.2, 2004.

[22] N. Sharma and U. Pal, *"Recognition of off-line handwritten devnagari characters using quadratic classifier,"* Comput. Vision, Graph. …, pp. 805–816, 2006.

[23] U. Bhattacharya, M. Shridhar, and S. K. Parui, *"OnRecognition of Handwritten,"* Comput. Vision, Graph.,817–828, 2006.

[24] M. A. Rahman and A. El Saddik, *"Modified syntactic method to recognize Bengali handwritten characters,"*IEEE Trans. Instrum. Meas., vol. 56, no. 6, pp. 2623– 2632, 2007.

[26] U. Pal, T. Wakabayashi, and F. Kimura, *"Handwritten bangla compound character recognition using gradient feature,"* Proc. - 10th Int. Conf. Inf. Technol. ICIT 20~~0~~A. Sharma, R. Kumar, and R. K. Sharma, *"Online Handwritten Gurmukhi Character Recognition Using Elastic Matching."* Image and Signal Processing, 2008. CISP'08. Congress on. Vol. 2. IEEE, 2008.

[27] D. Singh, M. Dutta, and S. H. Singh, *"Neural Network Based Handwritten Hindi Character Recognition System."* In Proceedings of the 2nd Bangalore Annual Compute Conference 2009.

[28] N. Das, B. Das, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri, *"Handwritten Bangla Basic and Compound character recognition using MLP and SVM classifier,"*J. Comput., vol. 2, no. 2, pp. 2151–9617, 2010.

[29] M. Kumar, *"Offline handwritten Gurmukhi character recognition : Study of different feature-classifier combinations,"* In Proceeding of the workshop on Document Analysis and Recognition, pp. 94-99, ACM, 2012.

[30] S. Iamsa-at and P. Horata, *"Handwritten Character Recognition Using Histograms of Oriented Gradient Features in Deep Learning of Artificial Neural Network,"* Int. Conf. IT Converg. Secur. 2013., no. 1, 1–5, 2013.

[31] E. Hassan, *"Exploiting multimedia content: A machine learning based approach,"* Electron. Lett. Comput. Vis. Image Anal., vol. 13, no. 2, p. 69, 2014.

[32] N. Dalal, B. Triggs, N. Dalal, B. Triggs, O. Gradients, and D. Cordelia, *"Histograms of Oriented Gradients for Human Detection*", Computer Vision and Pattern Recognition, (CVPR), IEEE Computer Society Conference on. Vol. 1. 2005.

[33] J. Arróspide, L. Salgado, and M. Camplani, *"Image-based on-road vehicle detection using cost-effective Histograms of Oriented Gradients,"* J. Vis. Commun. Image Represent., 2013.

[34] Q. Zhu, S. Avidan, M. C. Yeh, and K. T. Cheng, *"Fast Human Detection Using a Cascade of Histograms of Oriented Gradients,"* IEEE Conf. Comput. Vis. Pattern Recognit., vol. 2, pp. 1491–1498, 2006.

[35] A. Bosch, A. Zisserman, and X. Munoz, *"Representing shape with a spatial pyramid kernel,"* in Proceedings of the 6th ACM international conference on Image and video retrieval - CIVR '07, 2007.

[36] D. Salvi, J. Zhou, J. Waggoner, and S. Wang, *"Handwritten Text Segmentation using Average Longest Path Algorithm."* In Applications of Computer Vision (WACV), 2013 IEEE Workshop on, pp. 505-512. IEEE, 2013

[37] Y. Bai, L. Guo, L. Jin, and Q. Huang, *"A novel feature extraction method using pyramid histogram of orientation gradients for smile recognition,"* in Proceedings - International Conference on Image Processing, ICIP, 2009.


 

[38] C. Cortes and V. Vapnik, *"Support-Vector Networks,"* Mach. Learn., vol. 20, no. 3, pp. 273–297, 1995.
[39] V. N. Vapnik, *"An overview of statistical learning theory.,"* IEEE Trans. Neural Netw., vol. 10, no. 5, pp. 988–99, 1999.
[40] C.-C. Chang and C.-J. Lin, *"Libsvm,"* ACM Trans. Intell. Syst. Technol., vol. 2, no. 3, pp. 1–27, 2011.
[41] B. B. Chaudhuri and A. Majumdar, *"Curvelet-based multi SVM recognizer for offline handwritten bangla: Amajor Indian script,"* in Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, 2007.
[42] A. Dhurandhar, K. Shankarnarayanan, and R. Jawale, *"Robust pattern recognition scheme for devanagari script,"* in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2005

## Authors Profile

Hemant Yadav pursed Bachelor of Technology from UP Technical University,Lucknow in 2015 and pursuing Master of Technology from Indira Gandhi University ,Haryana in year 2019. His main research work focuses on Machine Learning, neural networks and Network security. He has more than 2 years of experience in IT industry.

Sapna Jain pursed Bachelor of Technology from UP Technical University,Lucknow in 2006 and Master of Technology from Teerthanker Mahaveer University ,Moradabad in year 2013. She is currently working as Assistant Professor in Department of Computer Science,Mata Raj Kaur Institute of Engineering & Technology, Rewari since 2009. She has published and presented more than 10 research papers in reputed international journals and conferences. Her main research work focuses on Network Security,Data Mining, IoT and Computational Intelligence based education. She has more than 8 years of teaching experience along with 2 year experience in IT industry.

 